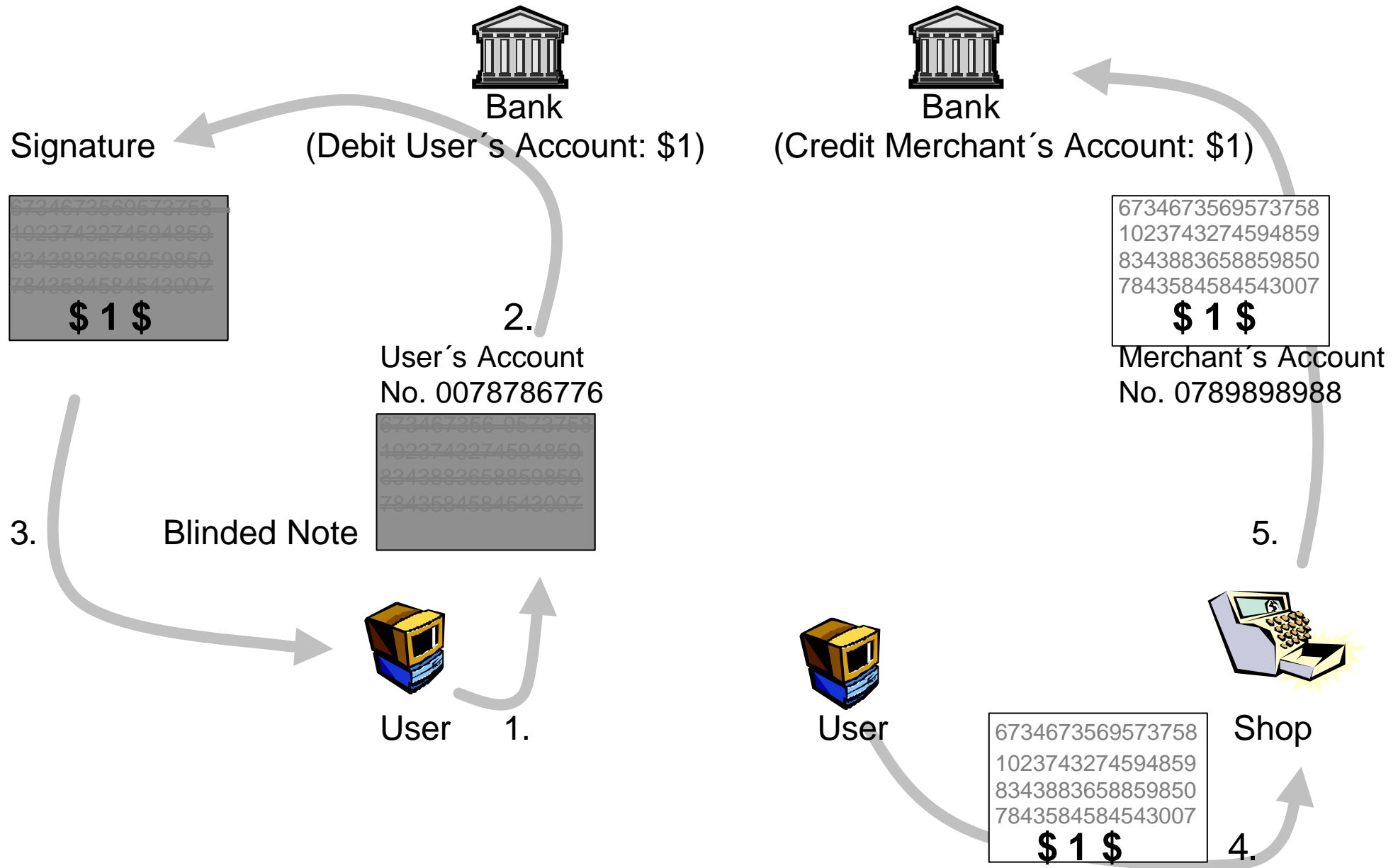


# Protecting User Identities at Application Level: ECash



# 8. Protecting User Identities at Application Level: Ecash by DigiCash [Chaum 1992]

## 8.1 Protocol steps for creating and spending untraceable electronic money:

### 1. Customer (Alice):

- generates a note number (100-digit number) at random
- in essence multiplies it by a blinding (random) factor
- signs the blinded number with a private key and sends it to the bank

### 2. Bank:

- verifies and removes Alice's signature
- debits Alice's account (by \$1)
- signs blinded note with a digital signature indicating its \$1-value and sends it to Alice

### 3. Customer (Alice):

- divides out the blinding factor
- uses bank notes (transfers it to shop)

### 4. Merchant (Bob):

- verifies bank's digital signature
- transmits note to bank

### 5. Bank:

- verifies its signature
- checks the note against a list of those already spent
- credits Bob's account
- sends signed "deposit slip" to Bob

### 6. Merchant (Bob):

- hands the merchandise to Alice together with his own signed receipt

## Blind signature scheme provides one-sided anonymity:

- Bank does not know blinding factor ->  
It cannot link note numbers that Bob deposits with Alice's withdrawals
- Anonymity of blinded notes through unpredictability of Alice's random numbers ->  
Alice can reveal random numbers and permit the notes to be stopped or traced

## 8.2 Mathematical protocol for issuing and spending untraceable money [Chaum et al. 1988]

$(e,n)$  : bank's public key

$(d,n)$ : bank's private key

1. Alice chooses at random  $x$  and  $r$ , and supplies the bank with :

$$B = r^e * f(x) \pmod{n}$$

where:             $x$ : serial number of bank note  
                      $r$ : blinding factor  
                      $f$ : one-way function

2. The bank returns  $B^d \pmod{n} = (r^e f(x))^d \pmod{n} =$   
 $r * f(x)^d \pmod{n}$

and withdraws one dollar from her account

3. Alice extracts  $C = B^d / r \pmod{n} = f(x)^d \pmod{n}$   
from  $B$

4. To pay Bob one dollar, Alice gives him the pair  
 $(x, f(x)^d \pmod{n})$

5. Bob immediately calls the bank, verifying that this note has not  
already been deposited

## Reason why one-way function $f$ is needed:

Suppose  $(x, x^d \bmod n)$  is electronic money

→ money can be forged:

1. choose  $y$
2. exhibit  $(y^e \bmod n, y)$

To forge money of the form  $(x, f(x)^d \bmod n)$ , you have to produce  $(f^{-1}(y^e) \bmod n, y)$ .

## 8.3 Blind Signatures and Perfect Crime [von Solms et al. 1992]:

### Step 1:

Open a bank account, receive Ecash account, kidnap the baby

### Step 2:

2.1 Choose a set of xs ( $x_1, x_2, \dots, x_p$ ) and a set of rs ( $r_1, r_2, \dots, r_p$ )

2.2 - Compute set  $B_j$  where  $B_j = r_j^e * f(x_j) \pmod n$

- Mail  $B_j$  to the authorities with the threat to kill the baby, if the following instructions are not complied with:

a) For all  $j$ , compute the set  $D_j = B_j^d \pmod n$

$$= r_j * f(x_j)^d \pmod n$$

b) Publish  $D_j$  in a newspaper

2.3 Buy the newspaper and compute  $C_j = D_j / r_j \pmod n = f(x_j)^d \pmod n$

→  $\{ (x_j, C_j) \}$  represents legal authorised and untraceable money

Note: Conditions are worse as in usual kidnapping cases:

- Police cannot register serial number of bank notes
- No physical contact needed (to transfer blackmailed money)

## 8.4 Chaum / Fiat / Noar Protocol for Offline Electronic Cash

$(e,n)$  : public key of the bank

$(d,n)$ : private key of the bank

$k$ : security parameter

$f,g$ : two-argument collision-free one-way functions

$u$ : Alice's bank account number

$v$ : counter for Alice's bank account

$\oplus$ : bitwise xor ,  $\parallel$ : concatenation

### Protocol to create electronic coins:

1. Alice chooses  $a_i, c_i, d_i$  and  $r_i$  ( $1 \leq i \leq k$ ) at random from the residue (mod  $n$ )

2. Alice forms and sends to the bank  $k$  blinded candidates:

$$B_i = r_i^e * f(x_i, y_i) \pmod{n} \quad (1 \leq i \leq k),$$

where

$$x_i = g(a_i, c_i) \text{ and } y_i = g(a_i \oplus (u \parallel (v+i)), d_i)$$

3. The bank chooses at random subset  $k/2$  of  $k$  blinded candidate indices  $R = \{ i_j \}, 1 \leq i \leq k, 1 \leq j \leq k/2$

$$\text{Assume } R = \{ k/2 + 1, \dots, k \}$$

4. Alice has to reveal  $r_i, a_i, c_i$  and  $d_i$  for all  $i \in R$ .  
The bank checks whether it can form  $B_i$ .

5. The bank gives Alice

$$\prod_{i \in R} B_i^d = \prod_{1 \leq i \leq k/2} B_i^d \pmod n$$

and charges her account one dollar,  
increments Alice's counter  $v$  by  $k$ .

6. Alice can extract the electronic coin

$$C = \prod_{1 \leq i \leq k/2} f(x_i, y_i)^d \pmod n$$

and increments her copy of the counter  $v$  by  $k$ .

## Protocol to spend money:

To pay Bob one dollar, Alice and Bob proceed as follows:

1. Alice sends  $C$  to Bob
2. Bob chooses random bit string  $z_1, z_2, \dots, z_{k/2}$
3. Alice responds as follows, for all  $1 \leq i \leq k/2$  :
  1. If  $z_i = 1$ , then Alice sends Bob  $a_i, c_i$  and  $y_i$ .
  2. If  $z_i = 0$ , then Alice sends Bob  $x_i, a_i \oplus (u \parallel (v+i))$  and  $d_i$
4. Bob verifies that  $C$  is of the proper form and that Alice's responses fit  $C$ .  
(Bob calculates  $f(x_i, y_i)$  and checks the bank's signature)
5. Bob later sends  $C$  and Alice's responses to the bank, which verifies their correctness and credits his account.

The bank must store  $C, z_1, \dots, z_k$  and  $a_i$  (for  $z_i = 1$ ) and  $a_i \oplus (u \parallel (v+i))$  (for  $z_i = 0$ )

## Detection of “double spending”:

If Alice uses the same coin C twice:

- High probability that two different shop keepers will send different values for at least one  $z_j$ .
- Bank knows both  $a_j$  and  $a_j \oplus (u \parallel (v+i))$ .  
Thus, bank can find out  $u$ .